

# Rapport de Projet de programmation web avancée

Par Alexis CARON et William HENRY

url du site : <https://otherside2710-vue-depository-10.glitch.me>

## Table des matières

<b>Table des matières.....</b>	<b>2</b>
<b>Introduction.....</b>	<b>3</b>
<b>Notre site.....</b>	<b>4</b>
<b>Problèmes rencontrés .....</b>	<b>8</b>

## Introduction

En programmation web avancée, il nous était demandé de travailler avec un Framework qui se nomme vue.js . Notre objectif était de réaliser un application web : un logiciel applicatif qui ressemble à un site web, disponible sur navigateur, comme un site internet. Étant donné que nous sommes tous deux intéressés par les ordinateurs en général et plus particulièrement leurs composants, nous avons décidé de faire un site qui listerait certains composants pour pc.



---

*Figure 1: Image d'illustration de Vue.js*

Le projet étant un site web complet, il devait comporter une partie Front-end (tout ce qui est lié à la partie client, c'est-à-dire la partie du site que voit un visiteur quand il se rend sur le site) ainsi qu'une partie Back-end (toute la partie serveur, qui réalise les opérations et transmet les données, invisible pour le client).

## Notre site

Le site que nous avons décidé de faire est donc un site de composants pour pc.

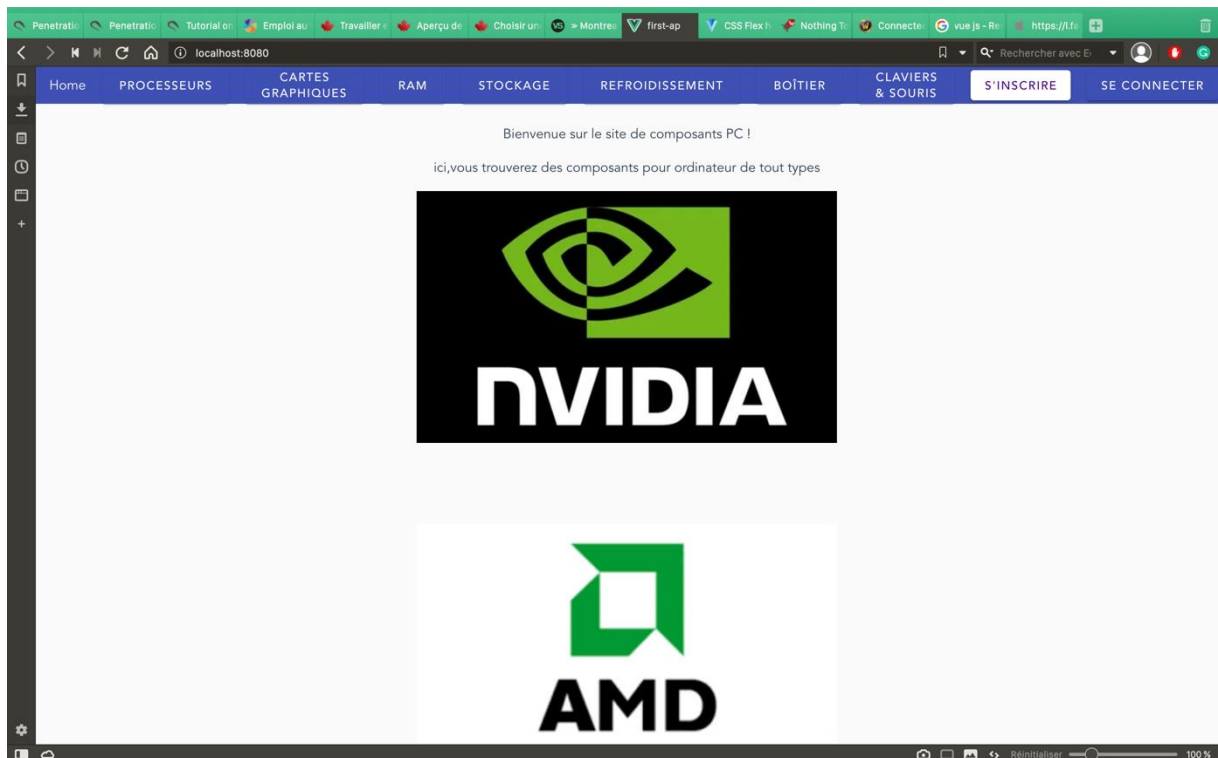


Figure 2: Page d'accueil de notre site de composants

Les différents boutons présents sur le site sont cliquables. Selon celui sur lequel on clique, l'url est changée, et un composant spécifique est chargé. Il est possible de faire un retour à la en cliquant sur « HOME ».

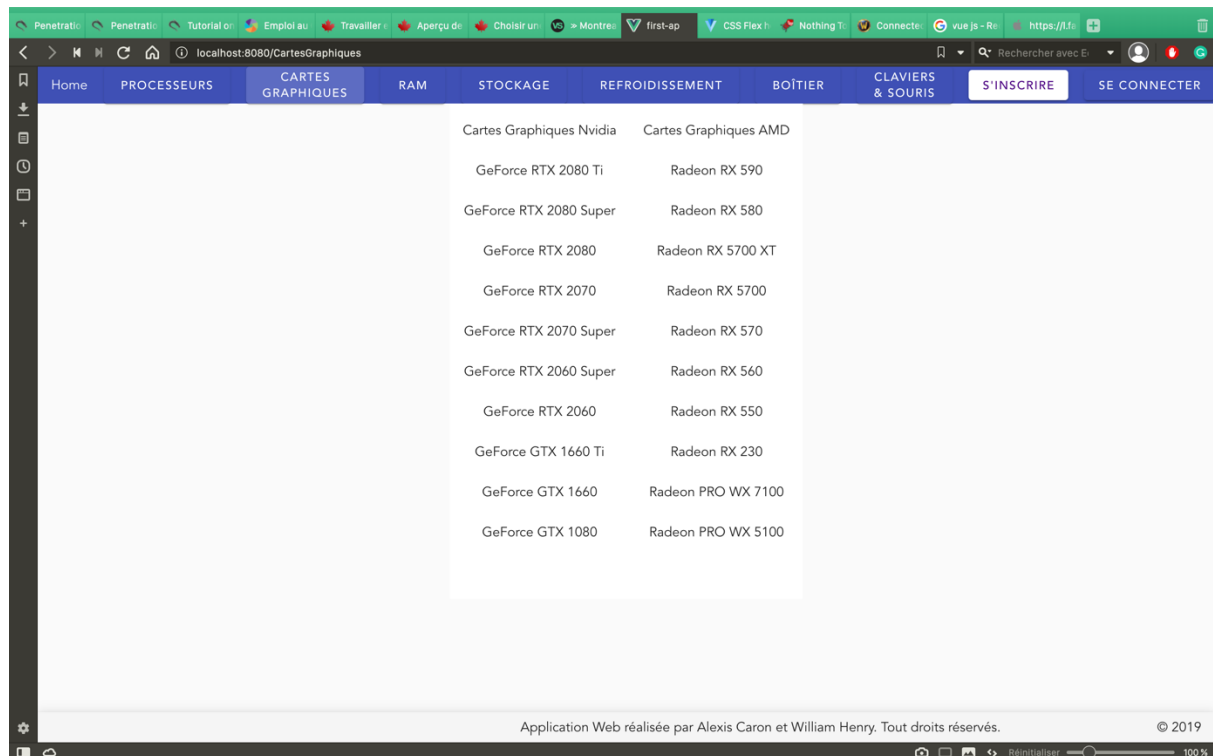


Figure 3: Exemple ou l'utilisateur à cliquer sur Cartes graphiques. On peut voir que l'url a bien été changé.

À noter que le site a été adapté en Responsive. S'il affiche les éléments en rangées sur la version web du site, tout a été modifié pour afficher les listes en colonnes si pour s'adapter à la taille de la fenêtre ou bien pour s'adapter à un Smartphone.

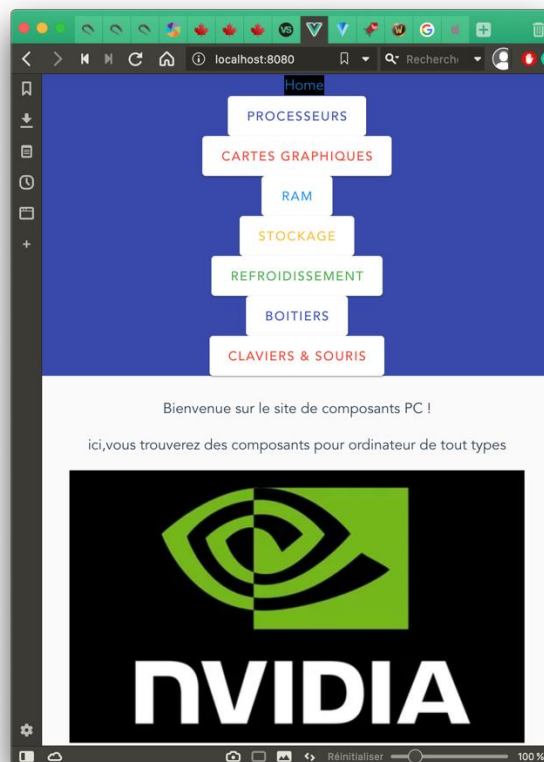


Figure 4: Page d'accueil du site sur une petite fenêtre

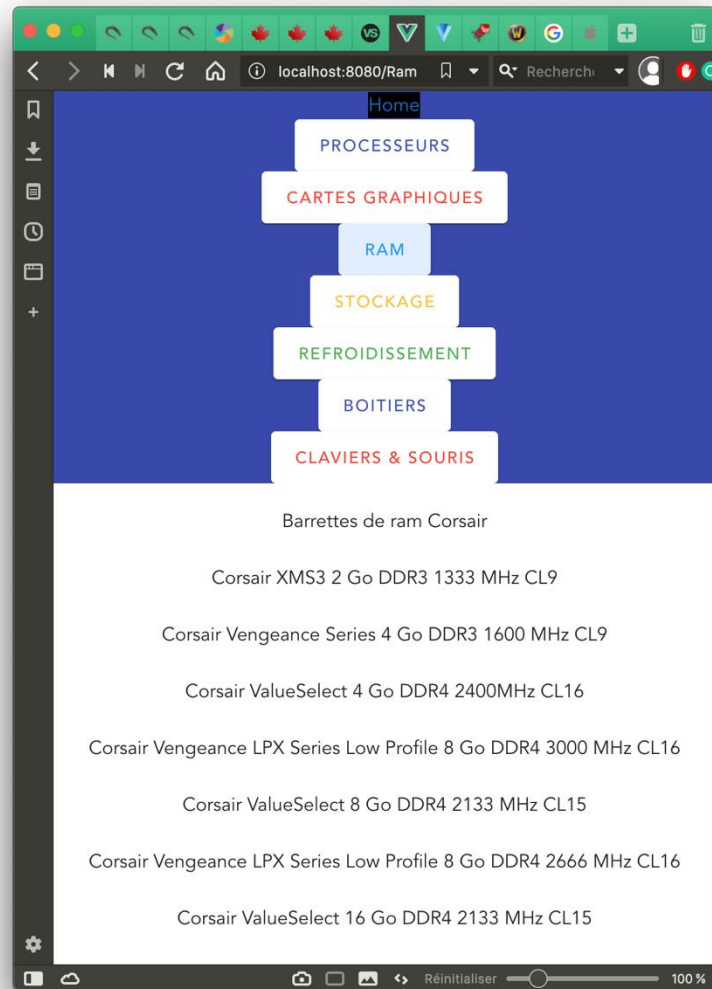


Figure 5 : exemple d'affichage avec la section RAM du site

Le site dispose également de formulaires d'inscription et de connexion.

Application Web réalisée par Alexis Caron et William Henry. Tout droits réservés. © 2019

Figure 6: Formulaire d'inscription du site

Application Web réalisée par Alexis Caron et William Henry. Tout droits réservés. © 2019

Figure 7: Formulaire de connexion

## Problèmes rencontrés

Durant la réalisation du projet, nous avons rencontrés de nombreux problèmes, dues au fait que ce projet était le premier que nous faisons en utilisant un Framework (jusqu'ici, nous n'avions travaillé qu'avec des fichier HTML/CSS, voire .js), et les débuts de l'utilisation de vue.js furent laborieux (nouvelles syntaxes, nouveau principes (exemples : composants)).

Nous avons donc dû nous habituer à l'utilisation de vue.js, car au début nous avons beaucoup de mal à faire des choses qui, en html par exemple, aurait été beaucoup plus simple à mettre en place, comme les routes par exemple. Ces dernières furent un énorme problème pour nous, car malgré de très nombreuses heures passées à regarder des tutoriels/cours sur internet, que nous avons reproduit exactement, l'url changeait bien, mais le composant affiché lui, ne bougeait pas. De plus, vue.js ne renvoyait pas d'erreur, et Visual studio code non plus, si bien que nous ne pouvions pas savoir déterminer d'où venait l'erreur. Nous avons finalement réussi à résoudre l'erreur simplement en déplaçant une balise dans notre App. Vue, et tout s'est mis à fonctionner d'un coup. Le souci est que ce problème nous a fait perdre énormément de temps. Nous avons également eue des soucis avec le design du site : en effet, sur certains écrans, les boutons de la version pc du site s'affichait correctement, alors que sur d'autres pc les boutons se chevauchaient, ce qui impactait énormément le design et l'ergonomie du site. Nous avons finalement réussi à résoudre le problème en se connectant au site depuis l'un des pcs qui avait ce problème (pour se mettre dans des conditions qui permettent de reproduire le bug afin de le corriger).

## Conclusion

En conclusion, nous sommes contents d'avoir réussi à faire ce que nous avons fait, même si nous n'avons pas réussi à faire la partie back-end (nous n'avons pas compris ce qui était attendu exactement pour cette partie-là), mais nous avons quand même réussi à faire entièrement la partie front-end, avec le responsive et le material design.